What is claimed is:

1. A method of efficiently storing content in a computing network, comprising steps of:

receiving hints regarding relationships among files; and

using the received hints to allocate storage for the files.

2. The method according to Claim 1, wherein the hints are created by a content management system.

3. The method according to Claim 1, wherein the hints specify one or more files that are likely to be referenced within a temporal proximity of a reference to a selected one of the files.

4. The method according to Claim 3, wherein the selected file is a web page.

5. The method according to Claim 4, wherein the one or more files comprise at least one of (1) one or more embedded objects of the web page and (2) one or more other web pages which are hyperlinked to the web page.

6. The method according to Claim 1, wherein the hints are created by a content authoring tool, and wherein the hints specify one or more files that are likely to be referenced within a temporal proximity of a reference to a selected one of the files.

7. The method according to Claim 6, wherein the selected file is a text document.

8. The method according to Claim 7, wherein the one or more files comprise one or more objects which are embedded within or referenced by the text document.

9. The method according to Claim 3, wherein the hints further specify weights which describe a degree of dependency for the relationships.

10. The method according to Claim 1, wherein the receiving step is performed by a file system and the using step is performed by a storage system.

11. The method according to Claim 2, wherein the hints are encoded in a markup language notation.

12. The method according to Claim 11, wherein the markup language notation is Extensible Markup Language ("XML") notation.

13. The method according to Claim 1, further comprising steps of:

receiving a request for one of the files;

retrieving the requested file from the allocated storage; and

returning the retrieved file.

14. The method according to Claim 1, further comprising steps of:

using the received hints to create dependency information which is stored by a receiver of the hints in temporary or permanent storage;

receiving a request for one of the files; and

determining a read request strategy for the requested file by accessing the stored dependency information.

15. The method according to Claim 14, wherein the read request strategy comprises determining selected ones of the files which should be pre-fetched along with a read of the requested file.

16. The method according to Claim 15, wherein the step of determining selected ones further comprises comparing a dependency weight of the files to a pre-fetch threshold.

17. The method according to Claim 16, wherein the pre-fetch threshold is used to tune the pre-fetch operation.

18. The method according to Claim 15, further comprising steps of:

retrieving the requested file from the allocated storage;

retrieving the selected ones from the allocated storage;

returning the retrieved requested file; and

caching the retrieved selected ones.

19. The method according to Claim 18, further comprising the step of caching the retrieved requested file.

20. A system for efficiently storing files in a computing network, comprising

means for receiving hints regarding relationships among files, wherein the hints specify one or more files that are likely to be referenced within a temporal proximity of a reference to a selected one of the files; and

means for using the received hints to allocate storage for the files.

21. A computer program product for efficiently storing files in a computing network, the computer program product embodied on one or more computer-readable media and comprising:

computer readable program code means for receiving hints regarding relationships among files, wherein the hints specify one or more files that are likely to be referenced within a temporal proximity of a reference to a selected one of the files; and

computer readable program code means for using the received hints to allocate storage for the files.